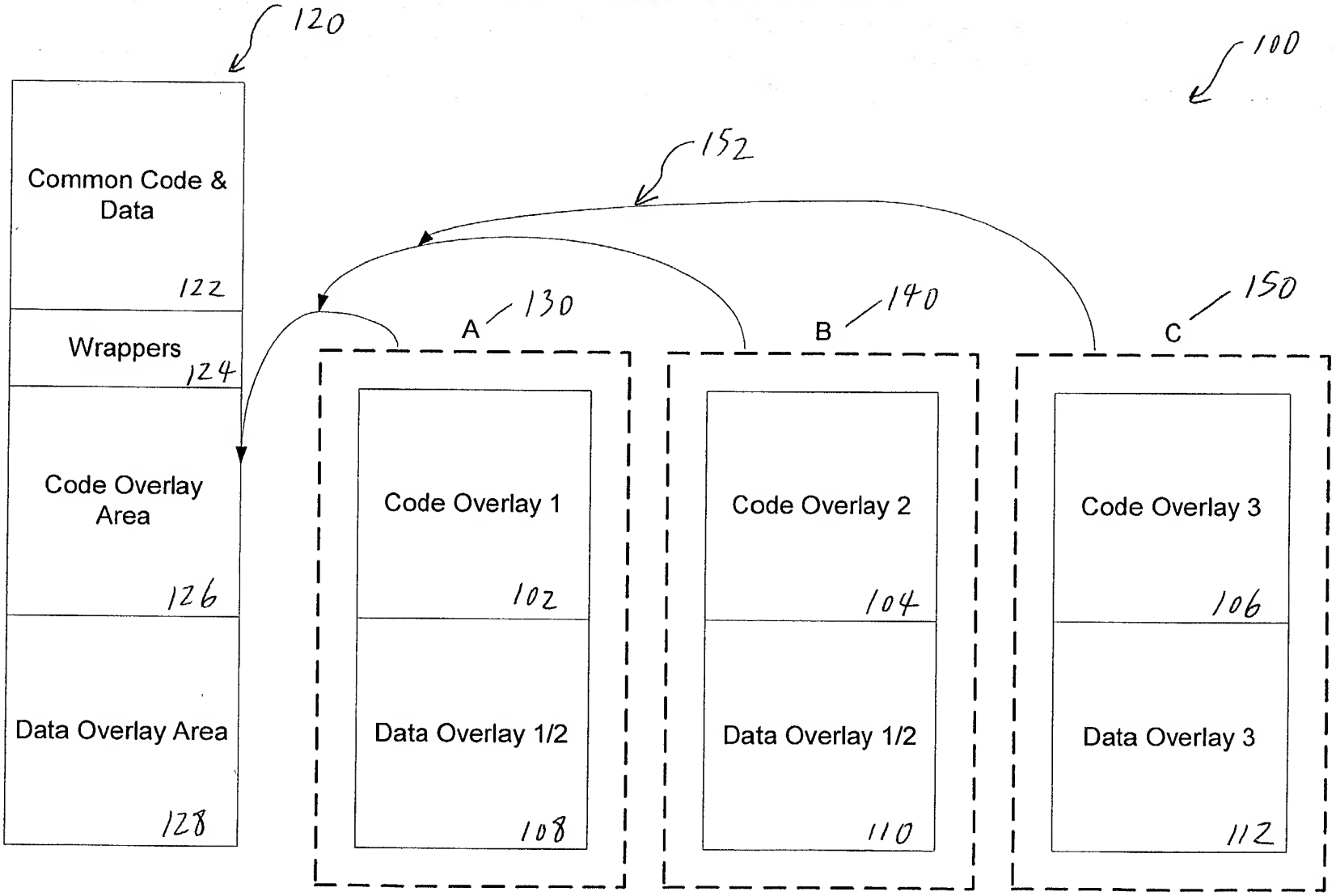
120
100
152
Common Code & Data
122
Wrappers
124
Code Overlay Area
126
Data Overlay Area
128
A
130
B
140
C
150
Code Overlay 1
102
Data Overlay 1/2
108
Code Overlay 2
104
Data Overlay 1/2
110
Code Overlay 3
106
Data Overlay 3
112

FIG. 1

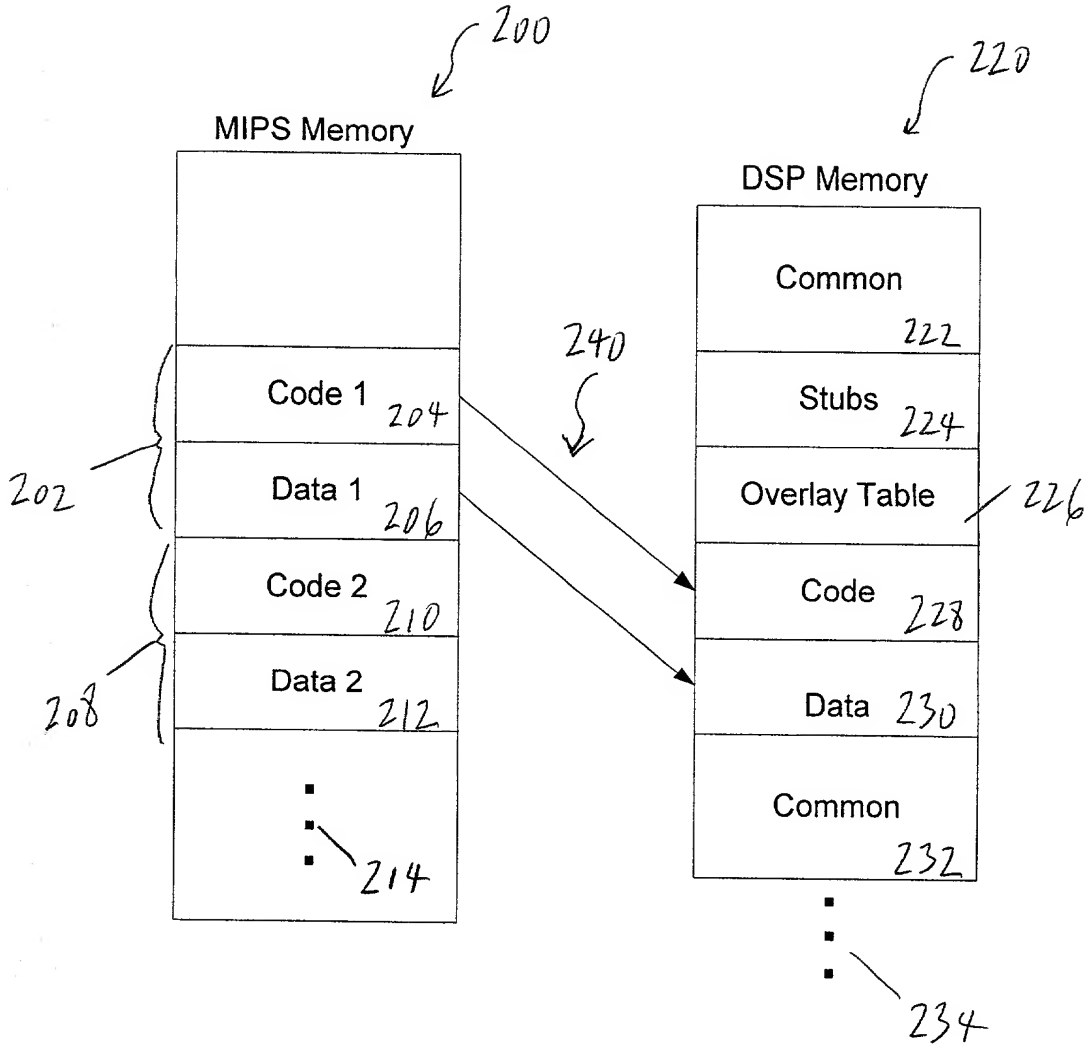
200
MIPS Memory
Code 1
204
Data 1
206
202
Code 2
210
Data 2
212
208
214
240
220
DSP Memory
Common
222
Stubs
224
Overlay Table
226
Code
228
Data
230
Common
232
234

FIG. 2

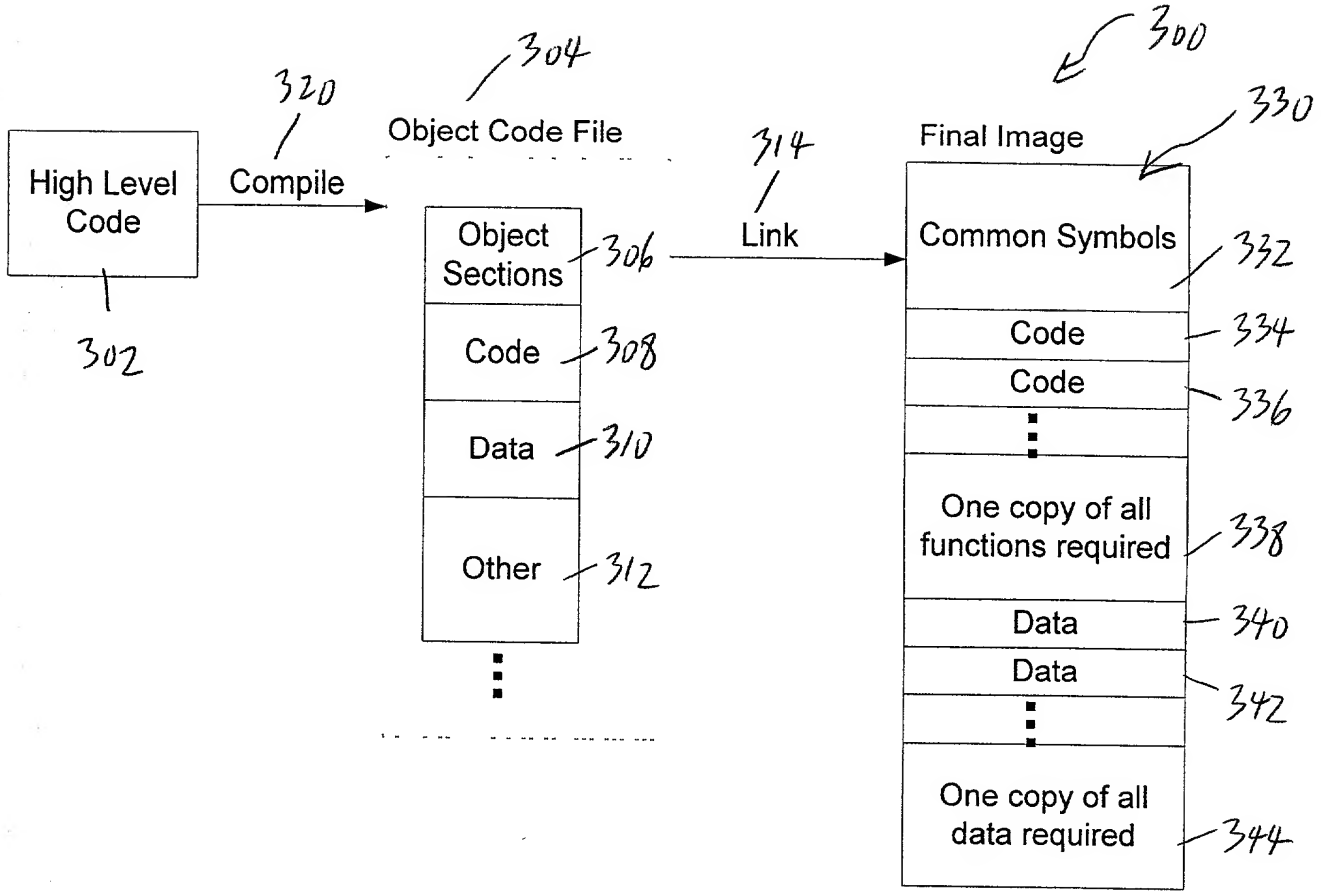

FIG. 3A (Prior Art)

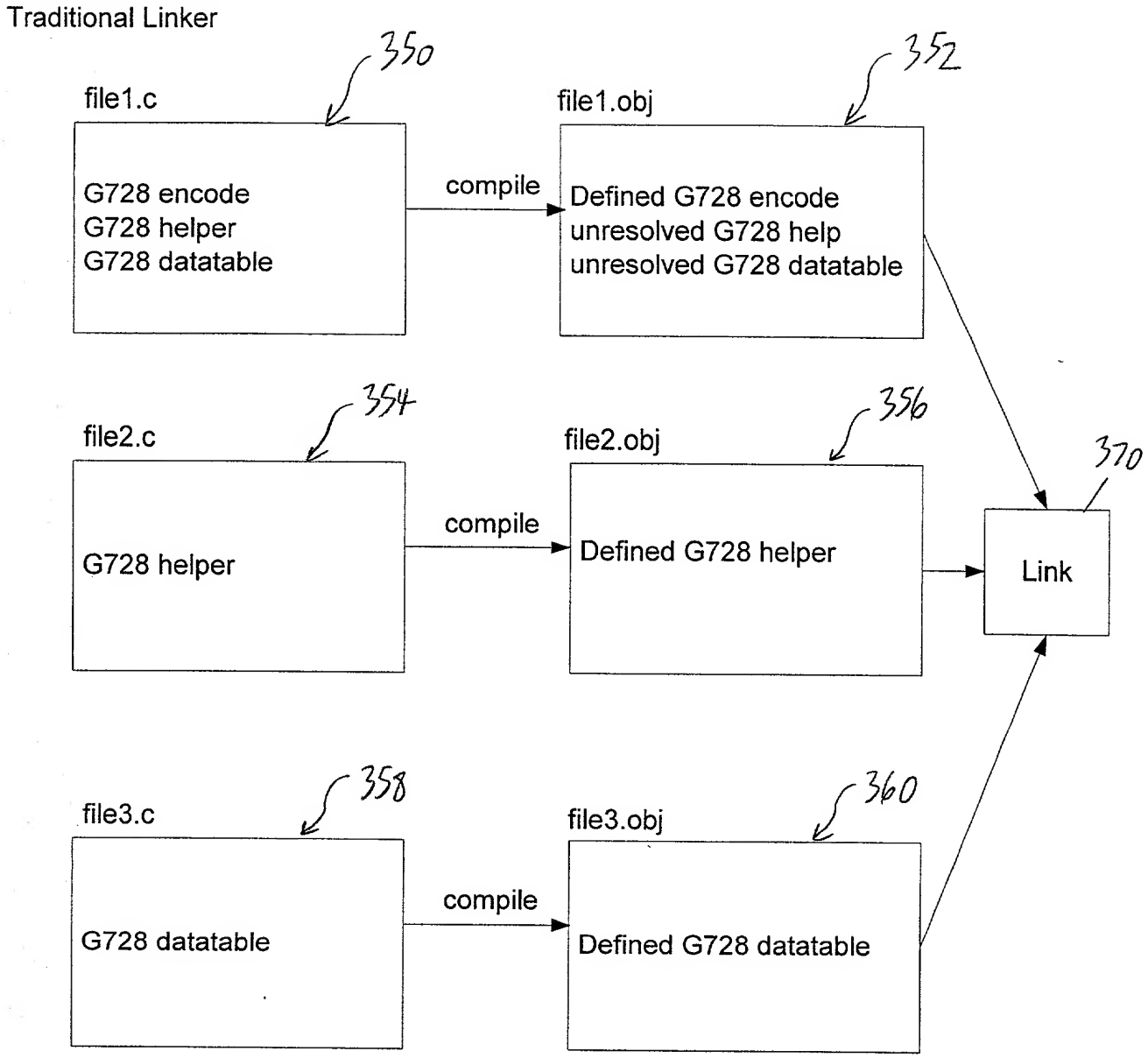
Traditional Linker
350
file1.c
G728 encode
G728 helper
G728 datatable
compile
352
file1.obj
Defined G728 encode
unresolved G728 help
unresolved G728 datatable
354
file2.c
G728 helper
compile
356
file2.obj
Defined G728 helper
370
Link
358
file3.c
G728 datatable
compile
360
file3.obj
Defined G728 datatable

FIG. 3B

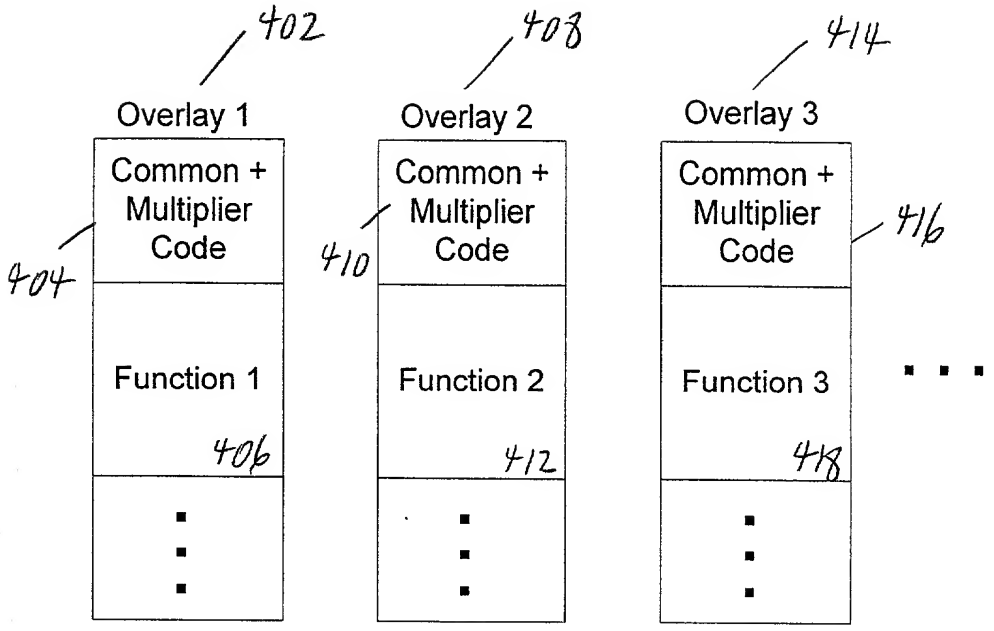

FIG. 4A (Prior Art)

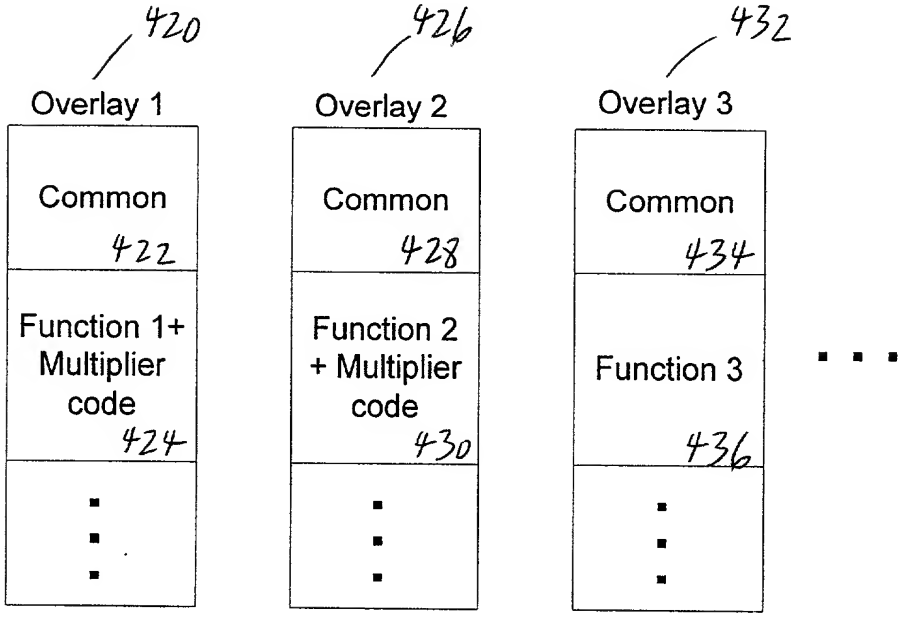

FIG. 4B

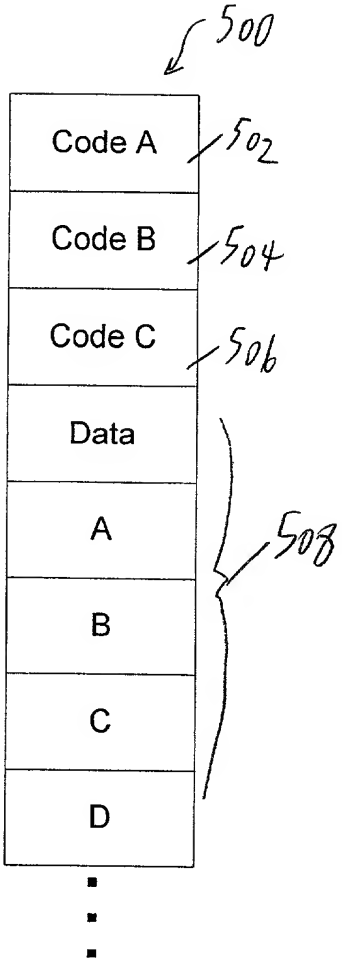

FIG. 5A (Prior Art)

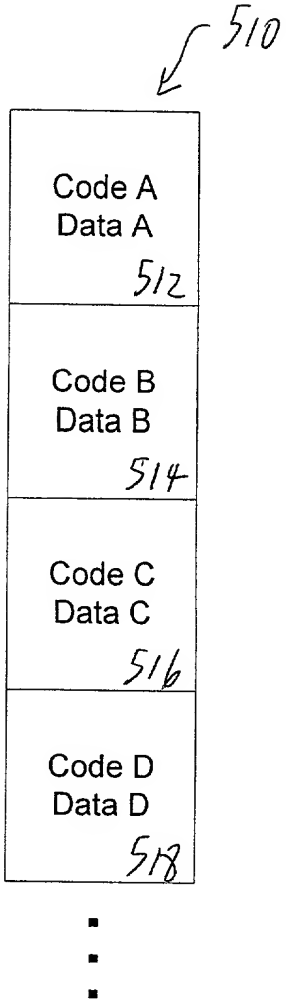

FIG. 5B

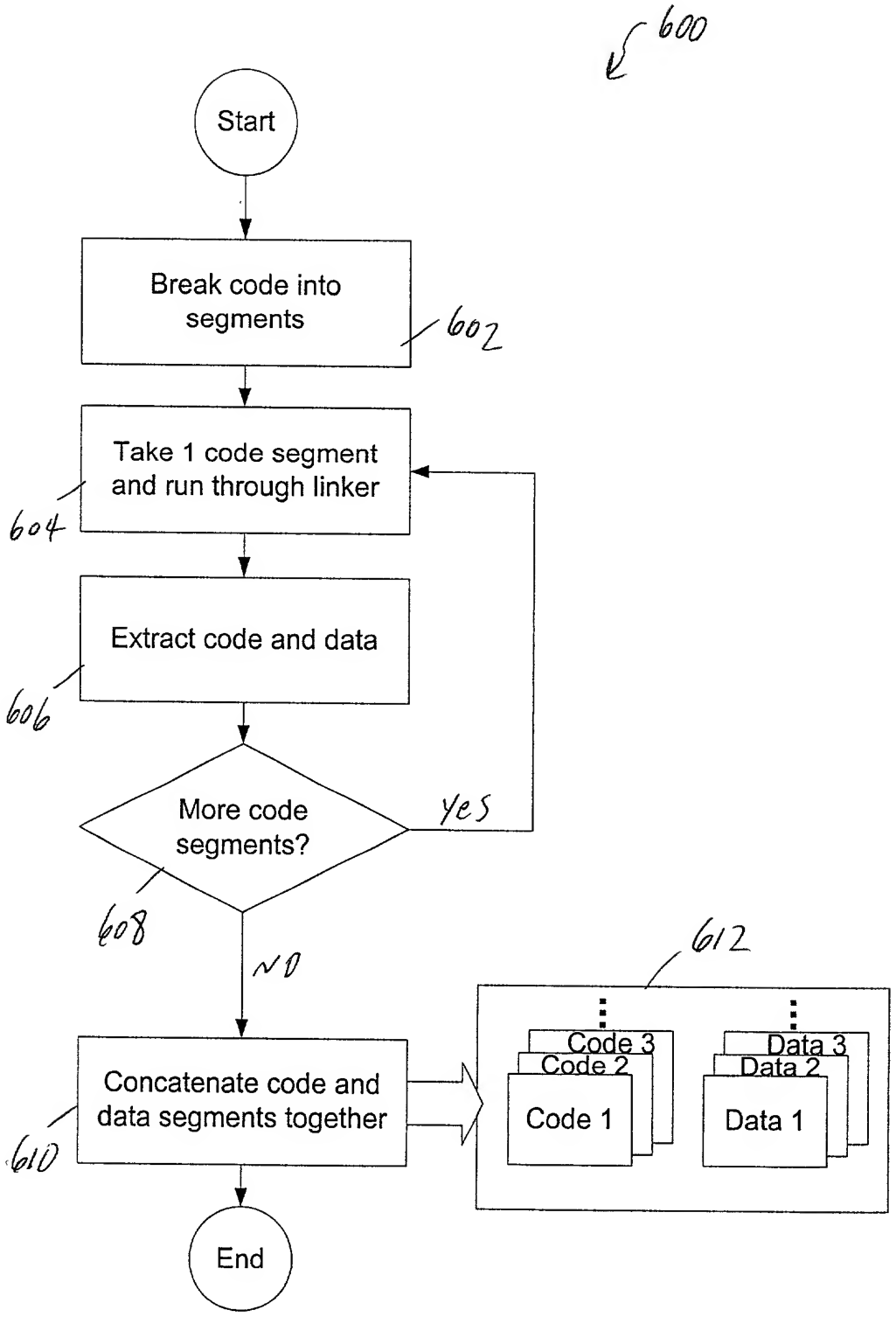
600
Start
Break code into segments
602
Take 1 code segment and run through linker
604
Extract code and data
606
More code segments?
yes
608
no
612
Concatenate code and data segments together
610
Code 3
Code 2
Code 1
Data 3
Data 2
Data 1
End

FIG. 6A

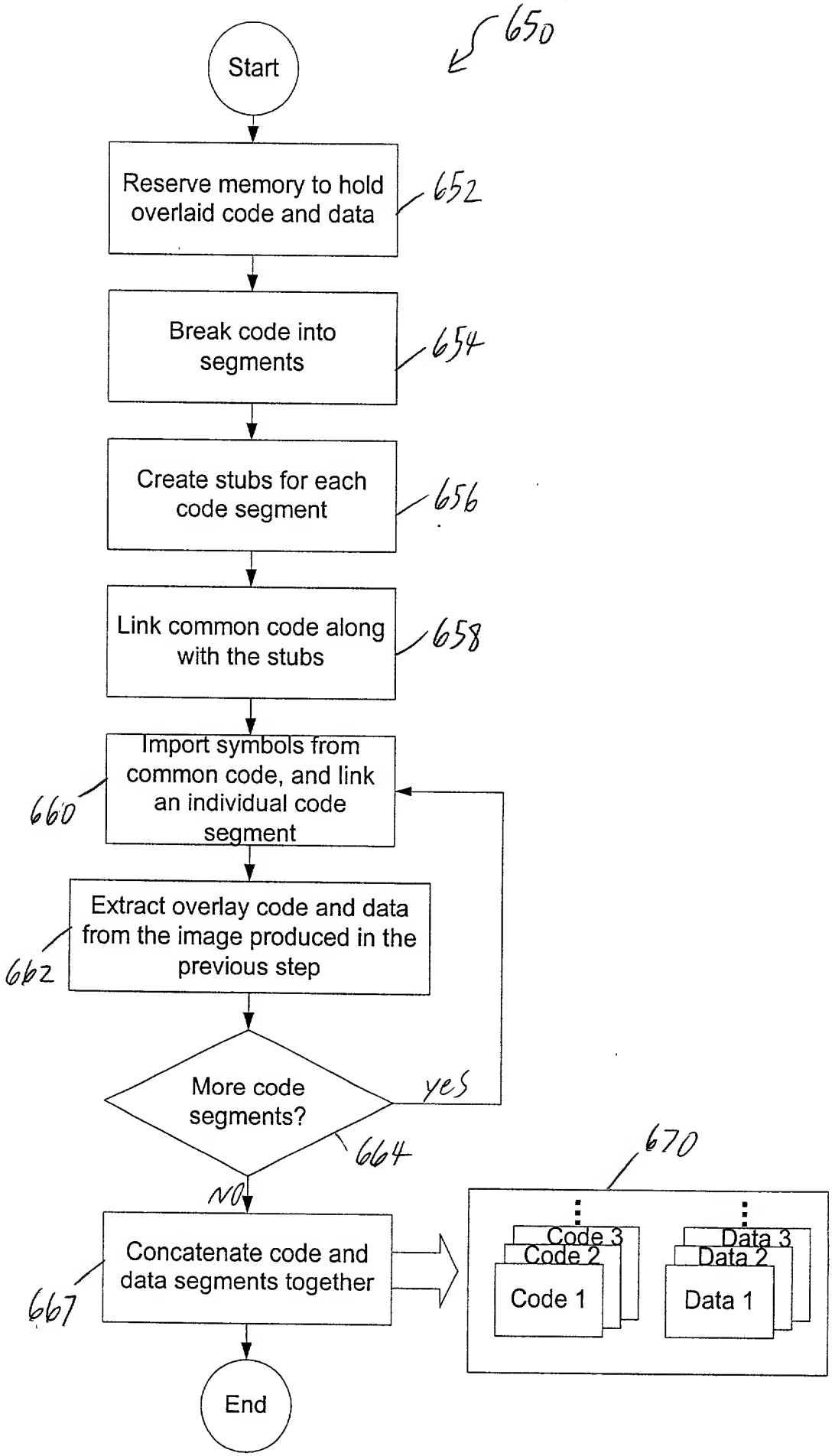
650
Start
Reserve memory to hold overlaid code and data
652
Break code into segments
654
Create stubs for each code segment
656
Link common code along with the stubs
658
Import symbols from common code, and link an individual code segment
660
Extract overlay code and data from the image produced in the previous step
662
More code segments?
yes
664
NO
Concatenate code and data segments together
667
670
Code 3
Code 2
Code 1
Data 3
Data 2
Data 1
End

FIG. 6B

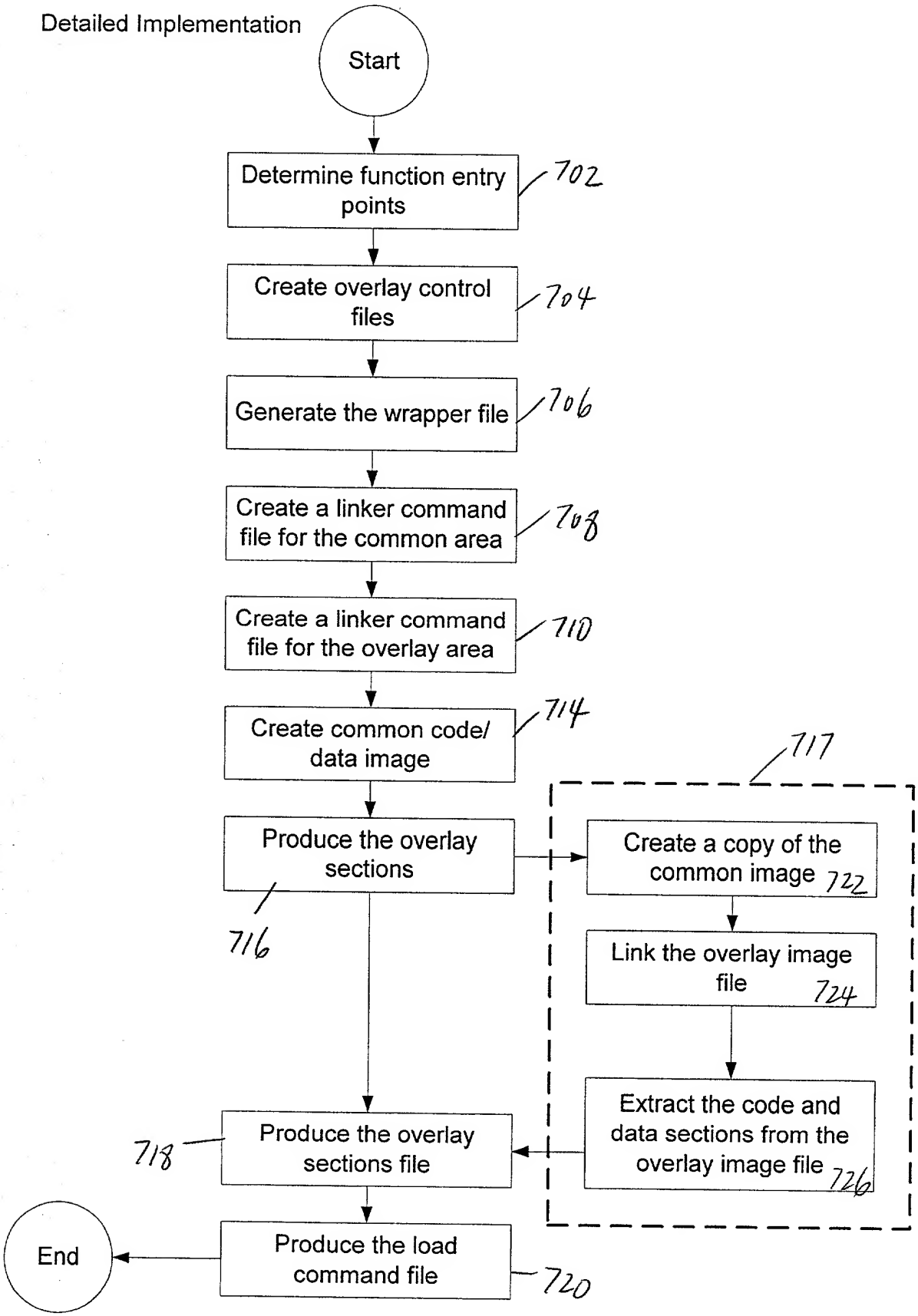
Detailed Implementation
Start
Determine function entry points
702
Create overlay control files
704
Generate the wrapper file
706
Create a linker command file for the common area
708
Create a linker command file for the overlay area
710
Create common code/ data image
714
717
Produce the overlay sections
716
Create a copy of the common image
722
Link the overlay image file
724
Extract the code and data sections from the overlay image file
726
718
Produce the overlay sections file
End
Produce the load command file
720

FIG. 7

Linker Command File:

Common Area

```
MEMORY
{
   . . . other memory sections removed . . .

   PAGEDCODE(X) : ORIGIN = 0x4200, LENGTH = 0x1800
   PAGEDDATA    : ORIGIN = 0x6000, LENGTH = 0x1400
}

SECTIONS
{
   . . . other sections removed . . .

   pagedCodeSection :
   {
      . = ALIGN(4);
      _dcm_gCodeOverlayAreaStart = ABSOLUTE( . );
      . = . + 0x1800 ;
      _dcm_gCodeOverlayAreaEnd = ABSOLUTE( . );
   } > PAGEDCODE

   pagedDataSect :
   {
      . = ALIGN(4);
      _dcm_gDataOverlayAreaStart = ABSOLUTE( . );
      . = . + 0x1400;
      _dcm_gDataOverlayAreaEnd = ABSOLUTE( . );

   } > PAGEDDATA

    .overlayDescr :
    {
       _dcm_gOverlayDescr = ABSOLUTE( . );
       *(overlay)

       SHORT( -2 )
    } > INTDATA
 }
```

FIG. 8A

Overlay Area

```
MEMORY
{
   . . . other memory sections removed . . .

   PAGEDCODE(X) : ORIGIN = 0x4200, LENGTH = 0x1800
   PAGEDDATA    : ORIGIN = 0x6000, LENGTH = 0x1400
}

SECTIONS
{
   . . . other sections removed . . .

   .text :
   {
      . = ALIGN(4);
      *(.text)
   } > PAGEDCODE

   .data :
   {
      . = ALIGN(4);
      *(.data)

   } > PAGEDDATA
}
```

FIG. 8B